

Amendments to the Claims:

The listing of claims supersedes all prior versions of claims in the application:

1. (currently amended) A method for providing customized provisioning of an application on a runtime environment of a terminal, the application including content having at least one content type, the method comprising the steps of:

obtaining the content by the runtime environment;

for each content type, obtaining by the runtime environment a set of provisioning instructions related to the content type, the provisioning instructions being customized by distributed provisioning control through the provisioning instructions for different versions of the application, the provisioning instructions embedded within content of the application for specifying a provisioning application program interface (API) set for provisioning the content ~~on~~ of the application terminal; and

executing by the runtime environment the provisioning instructions for employing the API set, by a script interpreter, to provision the application according to the specified content type.

2. (previously presented) The method according to claim 1, wherein provisioning control of the content is shared between the runtime environment and the application through the coupled provisioning instructions.

3. (original) The method according to claim 2 further comprising the step of employing a provisioning service to direct the provisioning API, the service configured for recognising the provisioning instructions.

4. (original) The method according to claim 3 further comprising the step of the service customizing the provisioning process and the associated provisioning API set according to the provisioning instructions.

5. (original) The method according to claim 4, wherein the service is shared by a plurality of the applications.
6. (original) The method according to claim 3 further comprising the step of employing a standard one of the provisioning API set by the service;
7. (original) The method according to claim 6 further comprising the step of obtaining remotely a custom API via a network coupled to the terminal.
8. (currently amended) The method according to claim 2, wherein the provisioning instructions are selected from the group comprising code, script, and configuration data[[],].
9. (cancelled)
10. (previously presented) The method according to claim 8, wherein the provisioning instructions are separate from the content.
11. (original) The method according to claim 10 further comprising the step of accessing the provisioning instructions remotely from the terminal.
12. (original) The method according to claim 11, wherein the remote access of the provisioning instructions is in conjunction with a networked repository.
13. (original) The method according to claim 12, wherein the terminal is selected from the group comprising wired devices and wireless devices.
14. (original) The method according to claim 5, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type.
15. (original) The method according to claim 14 further comprising the step of employing a series of enablers for providing access to corresponding selected ones of the generic API, each of the enablers associated with a predefined content type.

16. (original) The method according to claim 2, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type.

17. (original) The method according to claim 16 further comprising the step of employing a series of enablers for providing access to corresponding selected ones of the generic API, each of the enablers associated with a predefined content type.

18. (original) The method according to claim 17, wherein the enabler is an executable unit that executes provisioning instruction requests for the predefined content type.

19. (previously presented) The method according to claim 18 further comprising the step of obtaining the enabler selected from the group comprising: locally on the terminal by a provisioning service; bundled with a content descriptor of the content; and remotely from the terminal by the provisioning service.

20. (original) The method according to claim 5, wherein the provisioning instructions were amended prior to the step of obtaining the provisioning instructions by the runtime environment.

21. (previously presented) A terminal, including a computer processor and a computer readable storage medium for providing customized provisioning of an application on a runtime environment, the application including content having at least one specified content type, the terminal comprising:

a processing framework for obtaining the content;

a provisioning API set included in the processing framework for provisioning the content; and

a set of provisioning instructions related to the content, the provisioning instructions being customized by distributed provisioning control through the provisioning instructions for different versions of the application, the provisioning instructions embedded in content of the application for specifying selected ones of the provisioning API set; and

a script interpreter configured to employ the specified ones of the provisioning API sets to execute the provisioning instructions.

22. (previously presented) The terminal according to claim 21, wherein provisioning control of the content is shared between the framework and the application through the coupled provisioning instructions.

23. (original) The terminal according to claim 22 further comprising a provisioning service to direct the provisioning API, the service configured for recognising the provisioning instructions.

24. (previously presented) The terminal according to claim 23, wherein the service is configured for customizing the provisioning process and the associated provisioning API set according to the provisioning instructions.

25. (original) The terminal according to claim 24, wherein the service is shared by a plurality of the applications.

26. (currently amended) The terminal according to claim 23, wherein the service employs a standard one of the provisioning API set[[:]].

27. (previously presented) The terminal according to claim 26, wherein a custom API is obtained remotely by the service via a network coupled to the terminal.

28. (original) The terminal according to claim 22, wherein the provisioning instructions are selected from the group comprising code, script, and configuration data,

29. (cancelled)
30. (previously presented) The terminal according to claim 28, wherein the provisioning instructions are separate from the content.
31. (original) The terminal according to claim 30, wherein the provisioning instructions are configured for obtaining the remotely from the terminal.
32. (original) The terminal according to claim 31, wherein the remote access of the provisioning instructions is in conjunction with a networked repository.
33. (original) The terminal according to claim 32, wherein the terminal is selected from the group comprising wired devices and wireless devices.
34. (original) The terminal according to claim 25, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type.
35. (original) The terminal according to claim 34 further comprising a series of enablers for providing access to corresponding selected ones of the generic API, each of the enablers associated with a predefined content type.
36. (original) The terminal according to claim 22, wherein a generic API is included in the provisioning API set, the generic API configured for addressing by at least two dissimilar ones of the specified content type.
37. (original) The terminal according to claim 36 further comprising a series of enablers for providing access to corresponding selected ones of the generic API, each of the enablers associated with a predefined content type.
38. (original) The terminal according to claim 37, wherein the enabler is an executable unit that executes provisioning instruction requests for the predefined content type.

39. (previously presented) The terminal according to claim 38, wherein the enabler location is selected from the group comprising: locally on the terminal by a provisioning service; bundled with a content descriptor of the content; and remotely from the terminal by the provisioning service.

40. (original) The terminal according to claim 25, wherein the provisioning instructions were amended prior to the step of obtaining the provisioning instructions by the runtime environment.

41. (currently amended) A method for providing customized provisioning of an application on a runtime environment of a terminal, the application including content having at least one specified content type, the method comprising the steps of:

sending the content for receipt by the runtime environment;

for each content type, sending a set of provisioning instructions related to the content for receipt by the runtime environment, the provisioning instructions being customized by distributed provisioning control through the provisioning instructions for different versions of the application, the provisioning instructions embedded in content of the application for specifying a provisioning API set for provisioning the content of the application ~~on the terminal~~; and

making available selected ones of the API provisioning set for use by script interpreter;

wherein when executing the provisioning instructions, the provisioning service employs the API provisioning set to provision the application according to the specified content type.

42. (previously presented) A computer program product for providing customized provisioning of an application program on a runtime environment of a

terminal, the application including content having at least one specified content type, the computer program product comprising:

- a computer readable memory;

- a processing framework module stored on the computer readable memory for obtaining the content;

- a provisioning service module coupled to the framework module, the provisioning service module configured for utilising a provisioning API set for provisioning the content; and

- an interpreter module coupled to the framework module, the interpreter module configured for interpreting a set of provisioning instructions related to the content for execution by the provisioning service, the provisioning instructions being customized by distributed provisioning control through the provisioning instructions for different versions of the application, the provisioning instructions embedded in content of the application for specifying selected ones of the provisioning API set.